

LISTING OF CLAIMS

1. (Currently Amended) A computer-implemented method comprising:

identifying an empty node of a multiway radix tree data structure in a memory, the node at a node level of the multiway radix tree;

assigning a logical level number to a symbol in a key, the key being comprising a string of symbols, the symbol being a symbol in the key that corresponds to the node level of the empty node; and

storing an entry for the key in ~~a level of nodes of a multiway radix tree~~ the identified empty node, based at least in part on the logical level number instead of on a path between nodes of the multiway radix tree corresponding to ~~representing~~ every symbol in the key.

2. (Currently Amended) The method of claim 1, wherein the data structure ~~is comprised of levels of nodes~~ has node levels for the nodes arranged according to ascending magnitude of logical level numbers assigned to the ~~levels of nodes~~ node levels.

3. (Currently Amended) The method of claim 2, wherein a minimum number of symbols of the key are used to differentiate the key from other keys stored on the tree, and the data structure has only ~~levels of nodes~~ node levels for the symbols of the key that are used to differentiate the key from other keys stored on the tree.

4. (Currently Amended) The method of claim 2, wherein a search for the key uses the logical level number corresponding to a length of the key to find the ~~level of nodes storing~~ empty node where the key is stored.

5. (Currently Amended) A computer-implemented method comprising:

creating a root node and first-level nodes of a tree for storing keys in a memory, each key having a string of symbols;

BEST AVAILABLE COPY

creating a pointer from the root node to a node in the first-level of nodes corresponding to the first symbol in a key;

creating a second level of nodes to store an entry for the key in a node corresponding to the last symbol in the key; ~~and~~

creating a pointer from the node in the first-level of nodes corresponding to the first symbol in the key to the node in the second level of nodes corresponding to the last symbol in the ~~key~~ key;

assigning the first level of nodes a first logical level value of one, corresponding to the first symbol in the key; and

assigning a second logical level value of n to the logical level number for the second level of nodes corresponding to the last symbol in the key, wherein n equals the number of symbols in the key.

6. (Canceled)

7. (Currently Amended) The method of ~~claim 6~~ claim 5, wherein a new level of nodes to store an entry for a new key in a node corresponding to the last symbol in the new key is added to and/or inserted between the existing levels of nodes in the tree based on ordering all the levels of nodes in the tree in sequence according to ascending logical level numbers.

8. (Original) The method of claim 7, further comprising rearranging pointers that exist between a parent level of nodes and a child level of nodes when a level of nodes having a logical level number between the logical level numbers of the parent level of nodes and the child level of nodes is inserted between the parent level of nodes and the child level of nodes.

BEST AVAILABLE COPY

9. (Original) The method of claim 8, wherein the rearranging includes maintaining a logical path between nodes representing symbols in a key as the number of logical levels between nodes representing a first symbol in the key and a last symbol in the key changes.

10-11. (Canceled)

12. (Currently Amended) An apparatus, comprising:

a register to store an entry for a key, the key having a string of symbols, the entry to be stored in a tree data structure representing a table, the tree data structure organized in node levels, a first node level to have a root node, and a next node level to have a sub-node; and

a node generator to add a node to the tree data structure, the node to have a node level corresponding to a symbol in the key, to store the key entry based on the last symbol of the key, wherein the node is to be assigned a logical level number corresponding to ~~the~~ a length of the key instead of corresponding to the node level corresponding to the symbol; and

wherein the logical level number of the added node ~~storing the key~~ is used to find the key instead of a path between nodes corresponding to each symbol in the key.

13. (Currently Amended) The apparatus of claim 12, further comprising:

a receiver to receive keys; and

an inserter to insert an entry for each key to be stored into ~~the~~ a node created by the node generator.

14. (Currently Amended) The apparatus of claim 12, further comprising:

a finder to:

follow a pointer to a node corresponding to the first symbol of the key being searched for;

search the node corresponding to the first symbol of the key for a pointer to a node level ~~of nodes~~ having the logical level number as of the key being searched for; and

BEST AVAILABLE COPY

search for an entry for the key on the node level ~~of nodes~~ having the logical level number of the key being searched for.

15-17. (Canceled)

Application No.: 10/043,764
Attorney Docket No.: 42390.P12328

-5-

Examiner: Y. Wu
Art Unit: 2175